



Алгоритмы и способы их описания



- Алгоритм – это точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к искомому результату.
- То есть алгоритм – это четкое указание исполнителю алгоритма выполнить определенную последовательность действий для решения поставленной задачи и получения результата.

Основные особенности алгоритмов:

- Наличие ввода исходных данных.
- Наличие вывода результата выполнения алгоритма
- Алгоритм должен иметь дискретную структуру
- Однозначность
- Конечность – исполнение алгоритма должно закончиться за конечное число шагов.
- Корректность – алгоритм должен задавать правильное решение задачи.
- Массовость (общность) – алгоритм разрабатывается для решения некоторого класса задач, различающихся исходными данными.
- Эффективность – алгоритм должен выполняться за разумное конечное время.

Способы записи алгоритмов

Разработанный алгоритм может быть представлен несколькими способами:

- на естественном языке (словесная запись алгоритма);
- в виде блок-схем (графическая форма);
- на языке программирования.

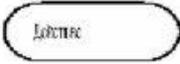
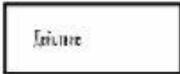
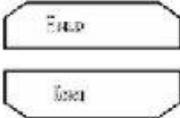
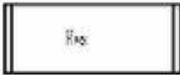
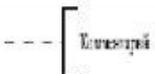
Правила составления блок-схем:

1. Каждая блок-схема должна иметь блок «Начало» и один блок «Конец».
2. «Начало» должно быть соединено с блоком «Конец» линиями потока по каждой из имеющихся на блок-схеме ветвей.
3. В блок-схеме не должно быть блоков, кроме блока «Конец», из которых не выходит линия потока, равно как и блоков, из которых управление передается «в никуда».
4. Блоки должны быть пронумерованы. Нумерация блоков осуществляется сверху вниз и слева направо, номер блока ставится вверху слева, в разрыве его начертания.
5. Блоки связываются между собой линиями потока, определяющими последовательность выполнения блоков. Линии потоков должны идти параллельно границам листа. Если линии идут справа налево или снизу вверх, то стрелки в конце линии обязательны, в противном случае их можно не ставить.

Правила составления блок-схем:

6. По отношению к блокам линии могут быть входящими и выходящими.
7. От блока «Начало» в отличие от всех остальных блоков линия потока только выходит, так как этот блок – первый в блок-схеме.
8. Блок «Конец» имеет только вход, так как это последний блок в блок-схеме.
9. Для простоты чтения желательно, чтобы линия потока входила в блок «Процесс» сверху, а выходила снизу.
10. Чтобы не загромождать блок-схему сложными пересекающимися линиями, линии потока можно разрывать. При этом в месте разрыва ставятся соединители, внутри которых указываются номера соединяемых блоков. В блок-схеме не должно быть разрывов, не помеченных соединителями.
11. Чтобы не загромождать блок, можно информацию о данных, об обозначениях переменных и т.п. размещать в комментариях к блоку.

СХЕМ АЛГОРИТМОВ ГОСТ ЕСПД 19.701-90

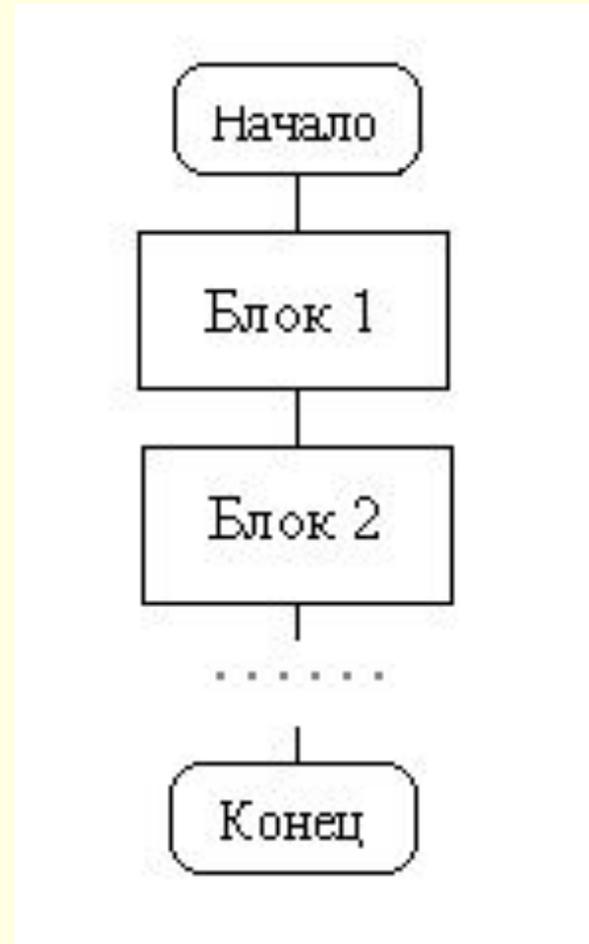
Название блока	Обозначение	Назначение блока
1. Терминатор		Начало, завершение программы или подпрограммы
2. Процесс		Обработка данных (вычисления, пересылки и т.п.)
3. Данные		Операции ввода-вывода
4. Решение		Ветвления, выбор, итерационные и поисковые циклы
5. Подготовка		Счетные циклы
6. Граница цикла		Любые циклы
7. Предопределенный процесс		Вызов процедур
8. Соединитель		Маркировка разрывов линий
9. Комментарий		Пояснения к операциям

Типы алгоритмов

- Линейные
- Разветвляющиеся
- Циклические

Линейный

- Линейный алгоритм состоит из упорядоченной последовательности действий, не зависящей от значений исходных данных, при этом каждая команда выполняется только один раз строго после той команды, которая ей предшествует.



Разветвляющиеся алгоритм

- Разветвляющимися называются алгоритмы, в которых в зависимости от значения какого-то выражения или от выполнения некоторого логического условия дальнейшие действия могут производиться по одному из нескольких направлений.

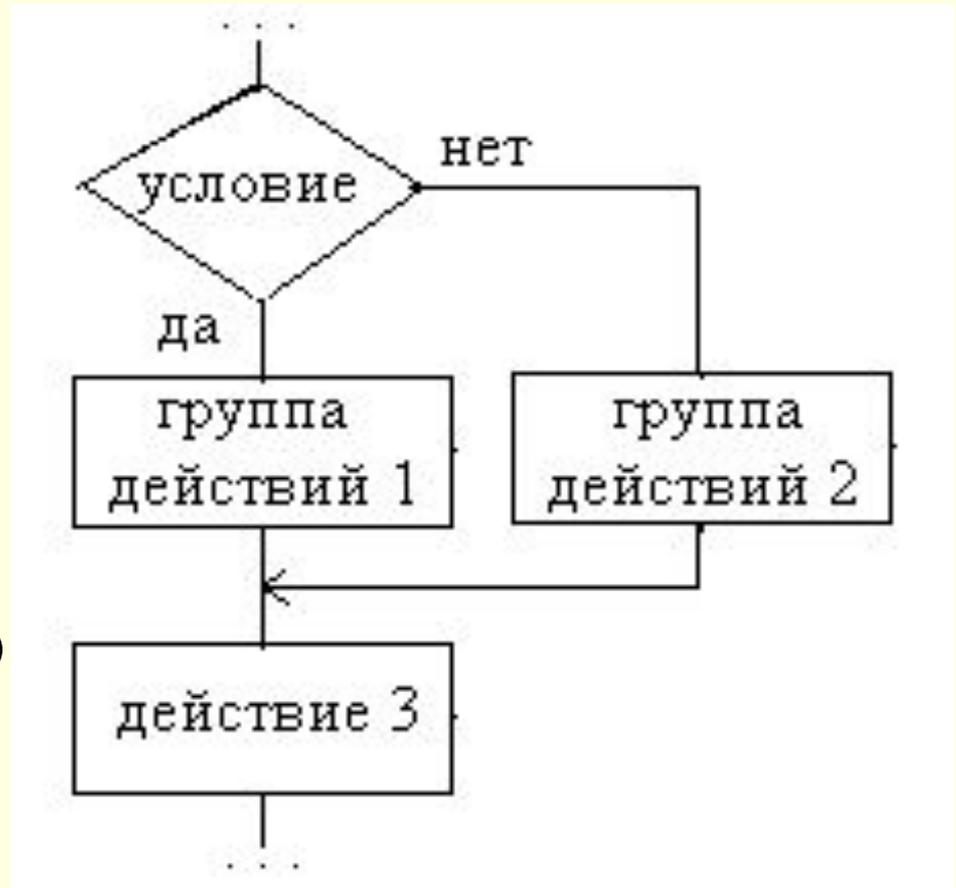
Виды разветвляющихся алгоритмов

1. «Обход» – такое разветвление, когда одна из ветвей не содержит ни одного оператора, т.е. как бы обходит несколько действий другой ветви.



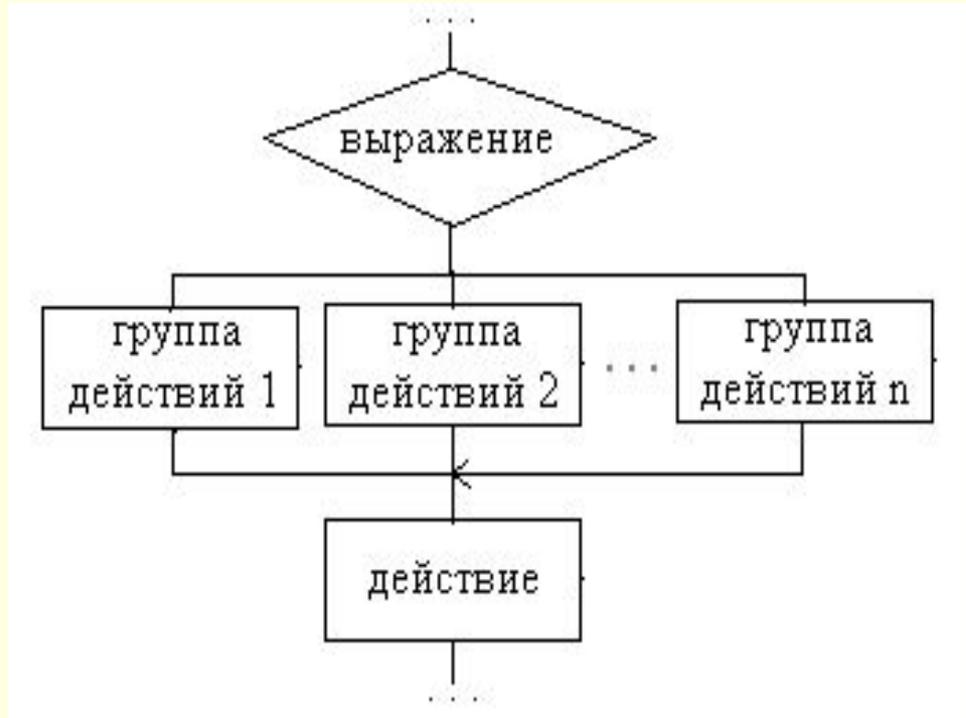
Виды разветвляющихся алгоритмов

2.
«Разветвление»
– такой тип
разветвления,
когда в каждой
из ветвей
содержится
некоторый набор
действий.



Виды разветвляющихся алгоритмов

3. «Множественный выбор» – особый тип разветвления, когда каждая из нескольких ветвей содержит некоторый набор действий. Выбор направления зависит от значения некоторого выражения.



Циклические алгоритмы

- Циклические алгоритмы применяются в тех случаях, когда требуется реализовать многократно повторяющиеся однотипные вычисления.

Цикл – это последовательность действий, которая может выполняться многократно, т.е. более одного раза.

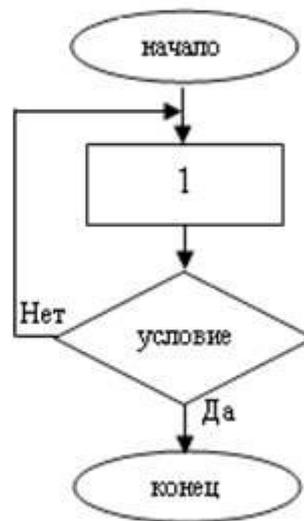
Циклические алгоритмы

- Различают:
 - циклы с известным числом повторений (или со счетчиком);
 - циклы с неизвестным числом повторений (циклы с предусловием и циклы с постусловием).

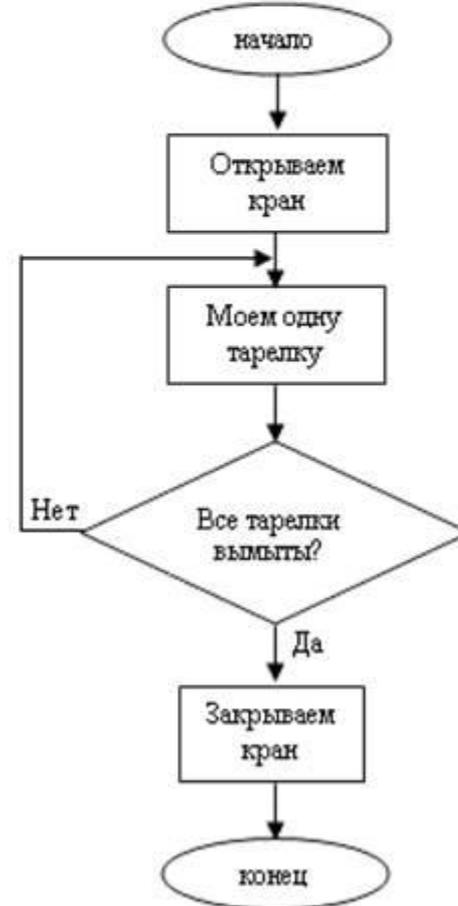
Циклический - блоки алгоритма выполняются до тех пор, пока не будет выполнено определенное условие.

Блок алгоритма 1 будет выполняться один или несколько раз до тех пор, пока не выполнится условие.

Общий вид:



Пример: Кран



Базовые алгоритмические структуры

https://www.youtube.com/watch?v=ct8agmyOW_U&feature=emb_title

7мин